

CMPT 215

Introduction to Computer Organization and Architecture

Assignment One

**Due Date: Friday January 27th, 6:00 pm – late submissions will not be accepted
all submissions MUST use the E-Handin system**

Total Marks: 82

1. (20 marks) Write, and test using SPIM, an assembly language program that prompts the user for a signed integer input. If the integer is smaller than 10, the program should output the string “Too small!”. If the integer is greater than 100, it should output the string “Too big!”. In either case, the program should then prompt the user for another input. If, however, the input integer is at least 10 but no more than 100, then the program should output the string “Just right!”, and terminate.
2. (20 marks) Write, and test using SPIM, an assembly language program that accepts as input a sequence of 10 signed integers. These numbers should be stored in an array in memory. Next, your program should prompt the user for an integer i . If this integer is between 1 and 10, your program should output the i 'th integer in the array, and then prompt the user for another integer i . Your program should terminate when the integer input by the user is outside the range of 1 to 10.
3. (20 marks) Write, and test using SPIM, an assembly language program that reads a character string, forms a new character string that contains the characters of the input string, but in reverse order, and then outputs the new string.
4. (12 marks) For each of the following pseudoinstructions, give a minimal length sequence of actual MIPS instructions to accomplish the same task. (Note: You will need to use \$at for some of the sequences. Also, note that the hexadecimal constant A000A₁₆ is too large to represent in just 16 bits, and you will need to use “lui”.)

(a) sub \$a0, \$a0, 3	(c) move \$s0, \$s1	(e) li \$s0, 0xA000A
(b) ble \$t7, \$t6, squeq	(d) li \$s0, 5	(f) addi \$t8, \$s0, 0xA000A
5. (10 marks) Outline the machine language instruction format (i.e., describe how the machine language instruction is divided into fields and state what each field is used for), and give the decimal value of each field, for each instruction in the following block of code. Assume that the code begins at memory address 5000₁₀. (Note: for the branch and jump instructions, you may find helpful the example on pages 98/99 of the text.)

```
loop:    beq $t8, $t5, done
          lw $s0, 12($s4)
          add $t8,$t8, $s0
          j loop
done:    andi $s1, $s0, 31
```

NOTE: For each of questions 1 through 3, turn in a separate file containing your program. Each program **must include comments** describing what exactly the program does, and how.